

ADO+ And XML Conceptual Overview

**Omri Gazitt
Group Program Manager
WebData
Microsoft Corporation**

2-232

Microsoft®
PDC 2000
Professional Developers Conference

Microsoft®
.net

the defining

point

Agenda

- **Motivation**
- **ADO+**
- **XML Framework**
- **Data and XML Together**
- **Demo**
- **Developer Roadmap**

Motivation

Q: Why another data access technology?

- **The New Landscape**
 - **The Internet changes everything**
 - **Loosely coupled architectures**
 - **XML emerging as universal data format**

The Problems

- **Fragmentation of programming models**
 - **Connected - relying on open connection to DBMS (OLE DB, ADODB)**
 - **Disconnected - request/response, no state between interactions (HTTP)**
- **Handling state is challenging**
- **Relational vs hierarchical data models**

ADO And XML Today

- **MDAC in Windows® 2000**
 - Based on 2-tier, extended for 3-tier (RDS)
 - Limited XML support
 - Object model lumps “connected” and “disconnected” together (RecordSet)
- **MSXML in Windows 2000**
 - Document-centric programming model
 - Integration with DBMS not easy
 - No stream level support

The Opportunity

- **.NET Framework requires something new**
- **Result: Data and XML Framework**
 - Natural evolution of ADO and MSXML
 - N-tier from the ground up
 - XML to the core
 - Unified, consistent data and XML architecture
 - Core part of .NET Frameworks

Agenda

- Motivation
- **ADO+**
- XML Framework
- Data and XML Together
- Demo
- Developer Roadmap

Introducing ADO+

- **Functional superset of ADO 2.6 (exception: server cursors)**
- **Contains the managed equivalent of OLE DB layer**
 - **No more COM/Automation dichotomy**
- **Explicit division between connected and disconnected object models**
 - **Managed Providers for connected access**

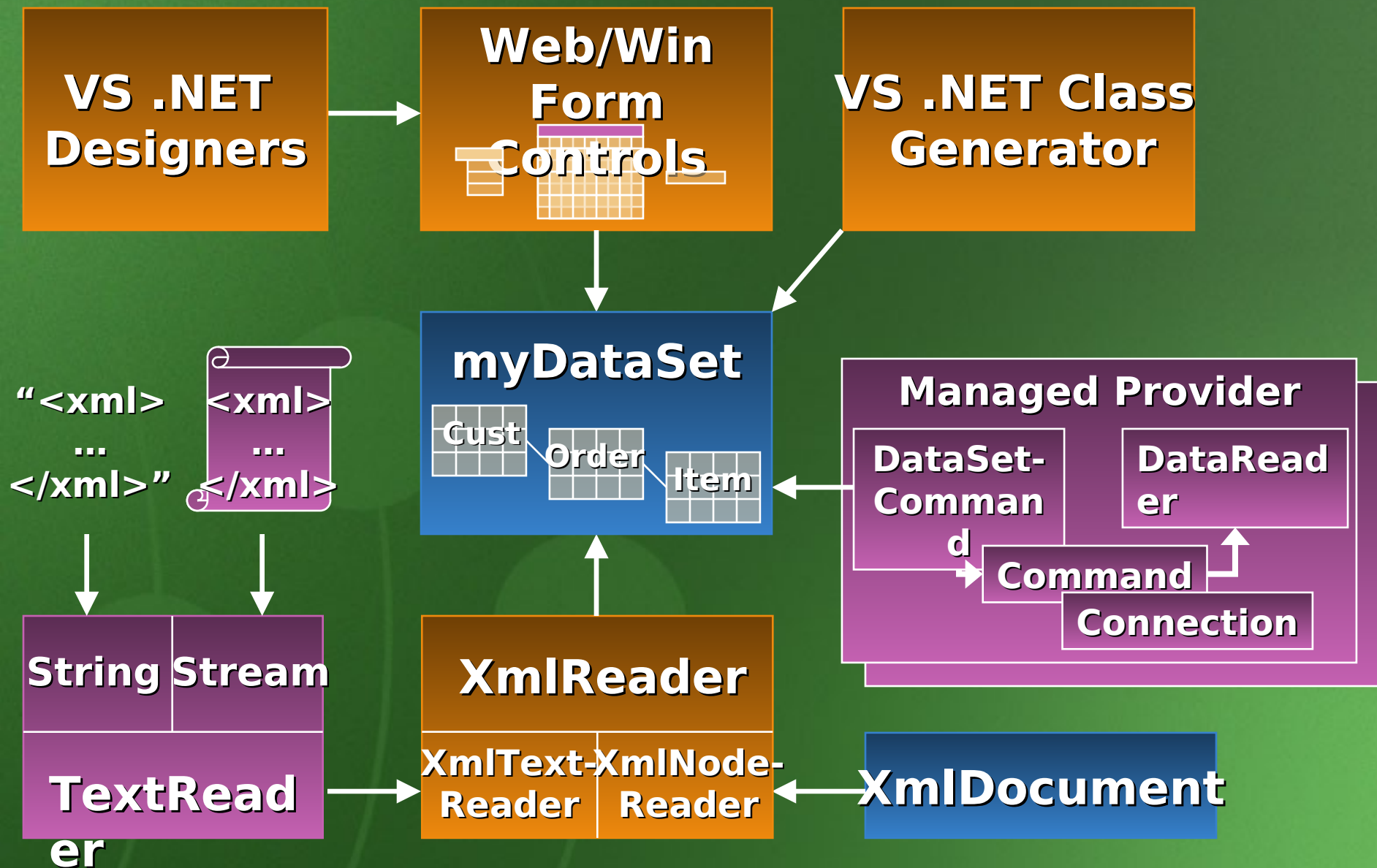
Managed Providers

- Object model similar to ADO
 - Connection, Command
- Efficient stream-level support for result-set processing
 - DataReader == Forward-only RecordSet
- New object to bridge the connected and disconnected layers
 - DataSetCommand - *Fill* a DataSet, *Update* back to the DB
- Two implementations: *SQL* and *ADO*

DataSet

- **Disconnected buffer of tables**
 - No knowledge of data source
 - Local sort/filter operations
- **Relational View of Data**
 - Tables, Columns, Rows, Relations
- **Great XML support**
 - Knows how to load/save XML (shreds)
- **Visual Studio .NET class-generation feature provides strongly-typed access**
 - `ds.Customers("John").Address`

ADO+ Architecture



ADO+ Benefits

- Much-improved disconnected story
- Great XML support in DataSet
- Language-neutral data access
- Great design-time integration with Visual Studio .NET
 - “Schema-specific” as well as ad-hoc data access
- Core part of .NET Framework
 - Expose same types, design patterns...

Agenda

- **Motivation**
- **ADO+**
- **XML Framework**
- **Data and XML Together**
- **Demo**
- **Developer Roadmap**

The XML Framework

- **Functional equivalent of MSXML 2.6**
 - VS .NET RTM == MSXML 3.0
- **Philosophy for XML components:**
 - Great standards support
 - Innovation where we think we can add value (programming model)
 - Chainable/pluggable architecture
 - Core part of .NET Framework

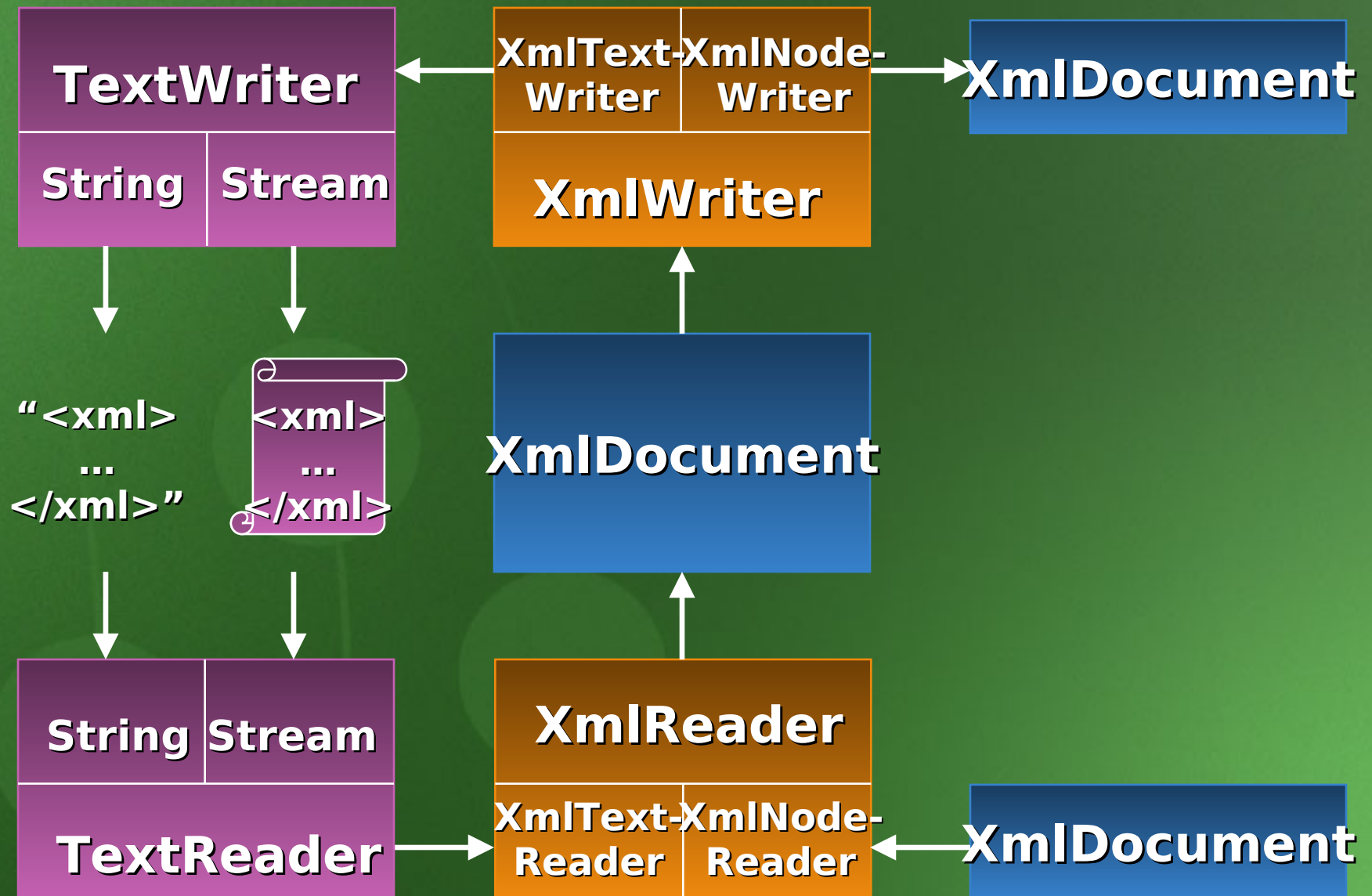
Standards Support

- DOM - all of level 1, some level 2
 - Exposed as XmlDocument class
- Validation: DTD, XDR (XSD beta2)
- XSL/T and X-Path (close to MSXML 2.6, complete in beta2)
- SAX ships as sample
- Others - SOAP, dig sig, W3C-QL...

Innovation

- XmlReader: stream-level “pull-model” parser
 - Can read a stream or document
 - Validation using a delegate (callback)
- XmlWriter: well-formed writer
 - Can write to a stream or document
- XmlNavigator: lens-oriented OM
 - X-Path support
 - Potential perf win
- XmlDataDocument: data-friendly DOM

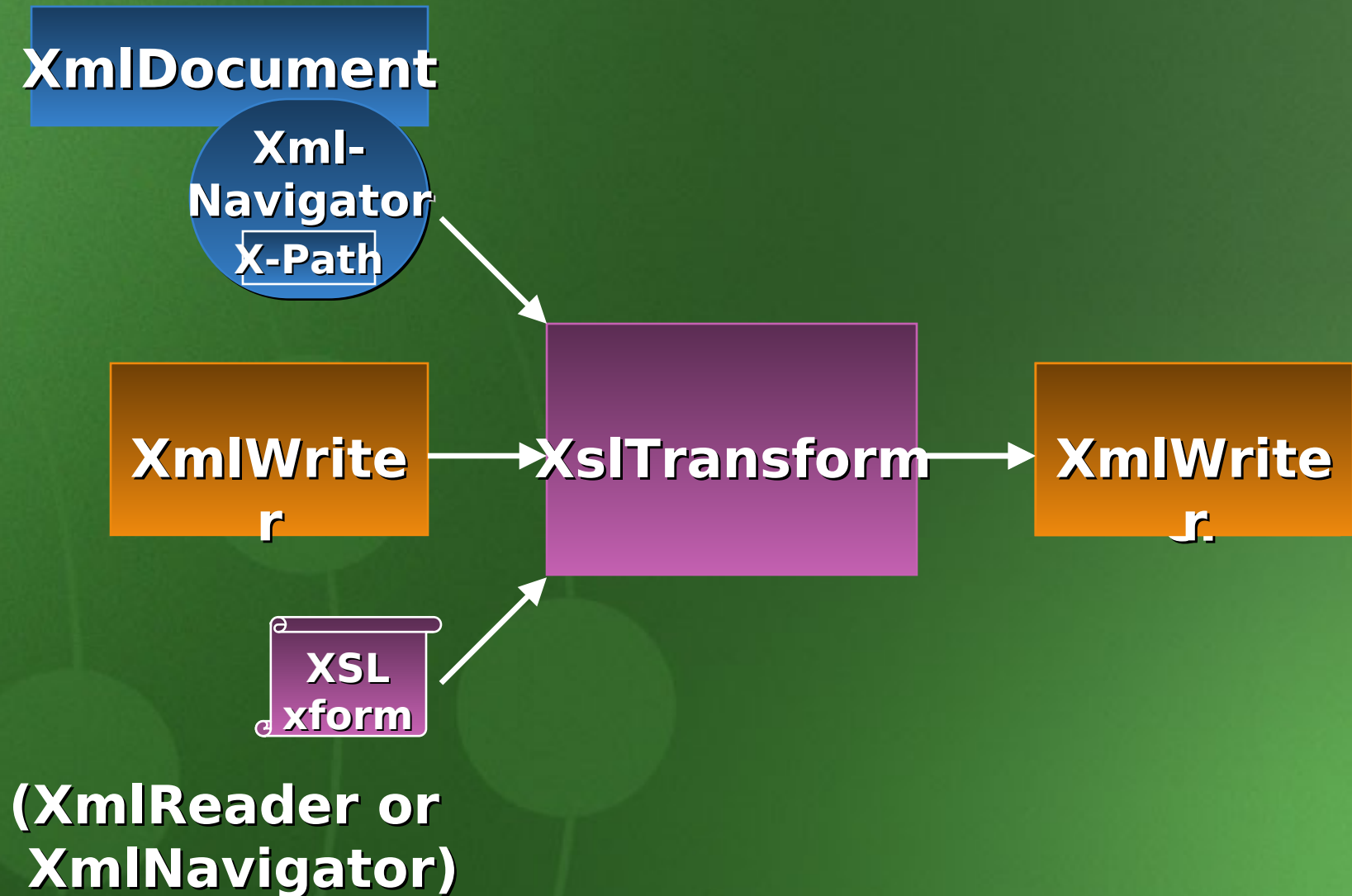
XML Architecture



XSL/T Support

- XSL/T engine binds to an XmlNavigator – pluggable
- Two built-in implementations:
 - DocumentNavigator – over XmlDocument
 - DataDocumentNavigator – over XmlDataDocument
- XSL/T produces an XmlReader or an XmlWriter
- Entire architecture is chainable

XSL/T Architecture



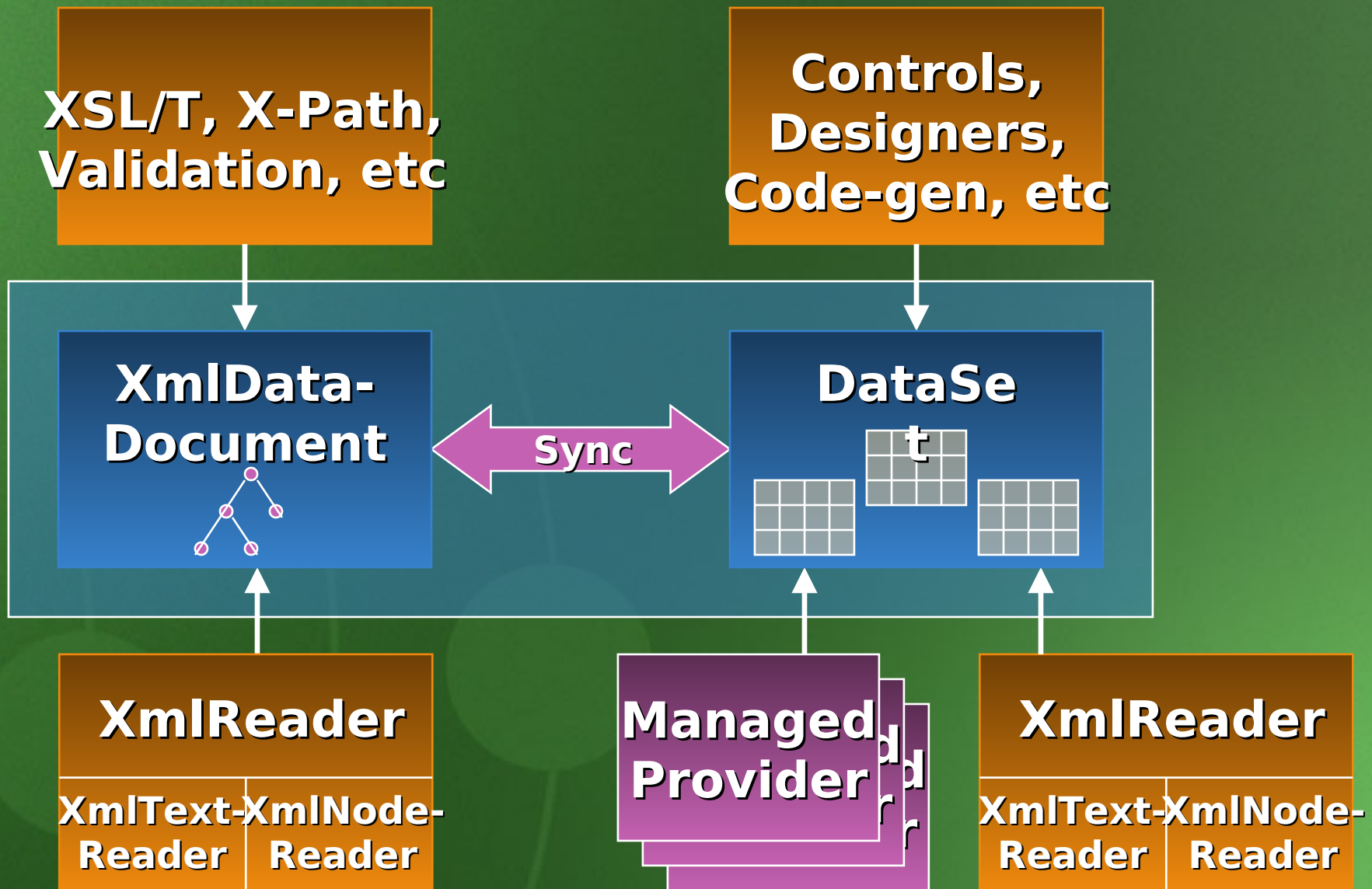
Agenda

- **Motivation**
- **ADO+**
- **XML Framework**
- **Data and XML Together**
- **Demo**
- **Developer Roadmap**

Data Is XML Is Data

- XmlDataDocument unifies XML and Data
- Plugs into entire stack of XML services
 - Subclass of XmlDocument - it's a DOM
 - Retains fidelity of underlying document
- Plugs into entire stack of Data services
 - Can obtain a DataSet view of an XmlDataDocument

Unified Architecture



Scenarios Enabled

- **Simultaneously expose data relationally or as XML**
 - Good for semi-structured data
 - Multiple tools on same data
- **Map interesting subset relationally using schema**
 - Use natural object model for each subset
- **Both DOM and DataSet can “grow up”**

Agenda

- **Motivation**
- **ADO+**
- **XML Framework**
- **Data and XML Together**
- **Demo**
- **Developer Roadmap**

Demo

- **Web site for order processing**
 - **Web Form with bound control (DataGrid) for displaying order items**
- **Read purchase order into XmlDocument**
- **Map order items (relational portion) to DataSet**
- **Modify items using DataGrid**
- **Transform document using**

Agenda

- **Motivation**
- **ADO+**
- **XML Framework**
- **Data and XML Together**
- **Demo**
- **Developer Roadmap**

Developer Roadmap

■ Data Consumers

- Existing ADO apps: **COM interop**
- Streaming scenarios, DML, singleton queries: **Managed Providers**
- Cache scenarios (sort, filter, bound controls): **DataSet**

■ Data Providers

- “Just data”: **expose as XML**
- Exposing a store: **OLE DB provider**
 - Wire into DQP, replication, DTS
- Legacy wire format: **Managed Provider**

Developer Roadmap

■ XML Consumers

- Interop with MSXML: **just use XML!**
- Streaming scenarios (e.g., message processing): **XmlReader**
- Caching scenarios (e.g., XSL/T): **XmlDocument**

■ Crossover Consumers (e.g., getting XML from SQL Server™ 2000)

- Completely relational XML: **DataSet**

Summary

ADO+ and XML Frameworks:

- **Natural evolution of ADO and MSXML**
- **N-tier from the ground up**
- **XML to the core**
- **Unified, consistent data and XML architecture**
- **Core part of .NET Framework**

Related Sessions

- **Other ADO+ and XML Framework talks**
 - Using ADO+ - Mike Pizzo/Tom Kaiser
 - Using the XML Framework - Chris Lovett
 - XML in Action - William Adams
 - Using Data and XML in ASP+ - Mike Pizzo
 - Data and XML in VS .NET - Sean Draine
- **Other WebData talks**

Where do **you** want to go today?

Microsoft